

Unix File System (UFS)

Indexed Allocation

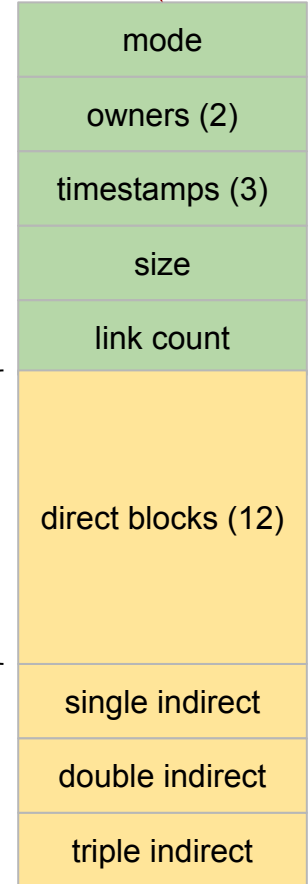
Filename	Index Block
one.txt	4
two.pdf	17

Disk Block #4

5
3
11
6
10
2

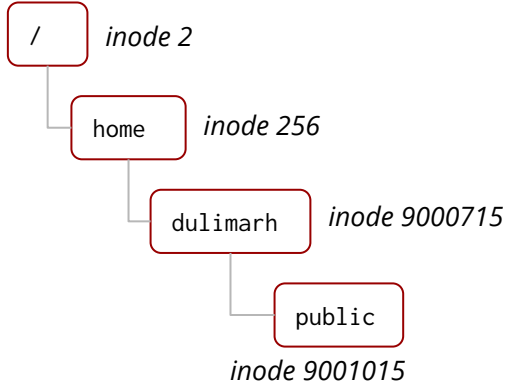
Filename	Inode Block
one.txt	241
two.pdf	1786

Unix inode



Pointers to data block

ONE inode per file/directory



. = self
.. = parent

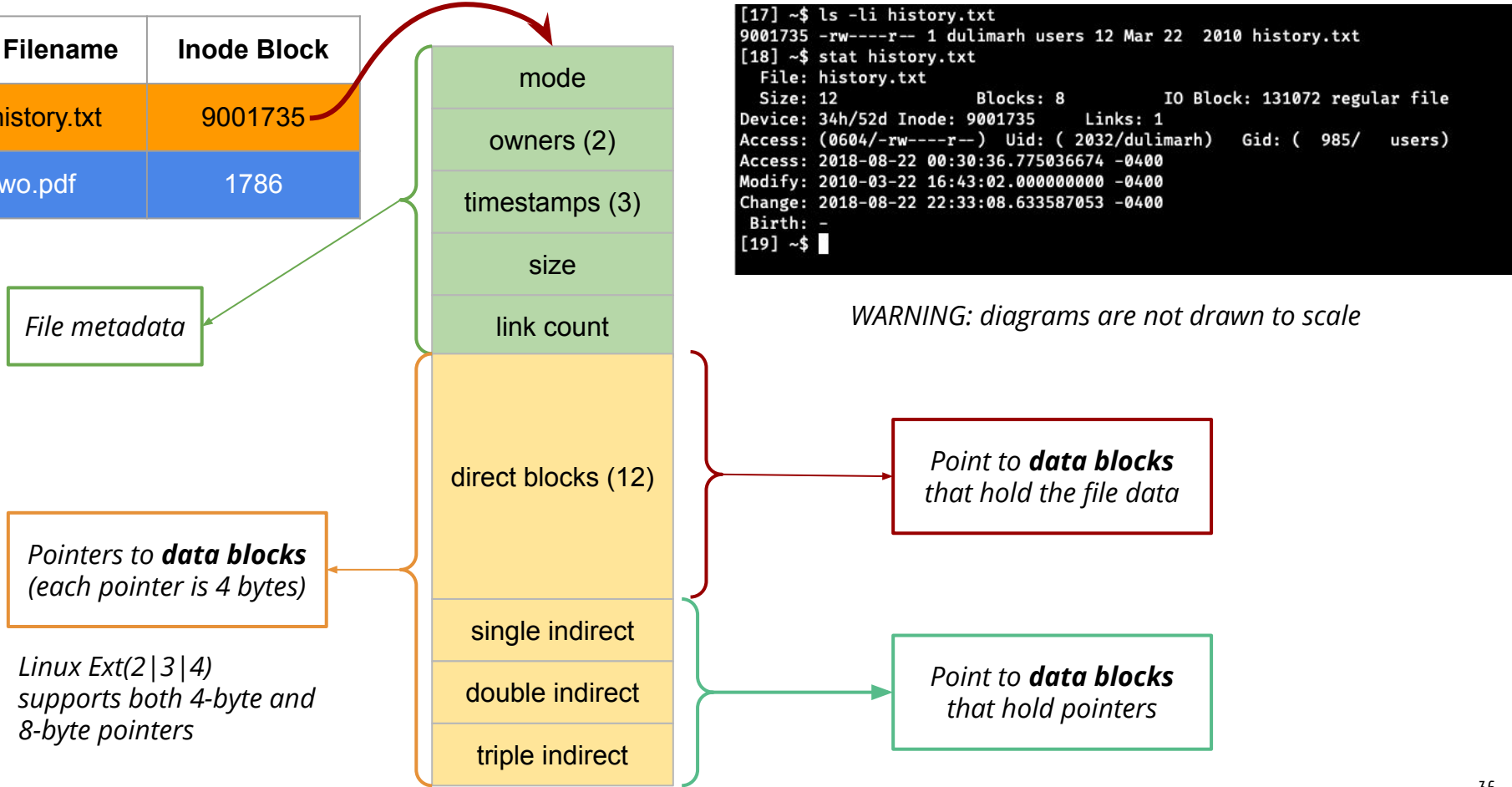
Absolute path: /home/dulimarh/public

```
# ls -l -a -i /
  2 drwxr-xr-x  18 root root    4096 Dec 19 10:39 .
  2 drwxr-xr-x  18 root root    4096 Dec 19 10:39 ..
256 drwxr-xr-x   1 root root   39316 Mar 27 10:09 home
# ls -l -a -i /home
 256 .
   2 ..
9000715 dulimarh
# ls -l -a -i /home/dulimarh
9000715 drwx----x  1 dulimarh users    4536 Apr  9 10:39 .
  256 drwxr-xr-x  1 root    root    39316 Mar 27 10:09 ..
9001001 drwx-----  1 dulimarh users     236 Apr  9 10:00 courses
9001015 drwxr-xr-x  1 dulimarh users     74 Jul 14 2014 public
9001017 drwx---r-x  1 dulimarh users    1416 Apr  9 10:33 public_html
# ls -l -a -i /home/dulimarh/public
total 68
 9001015 drwxr-xr-x  1 dulimarh users     74 Jul 14 2014 .
 9000715 drwx----x  1 dulimarh users   4536 Apr  9 10:39 ..
10075698 -rwxr--r-x  1 dulimarh users  61781 Feb 25 2014 Recursion.pdf
10075738 -rw-r--r--  1 dulimarh users   650 Feb 10 2010 StandardIO.java
10071587 drwxr-xr-x  1 dulimarh users   346 Apr  9 10:45 UnixHumor
```

Unix inode blocks and data blocks

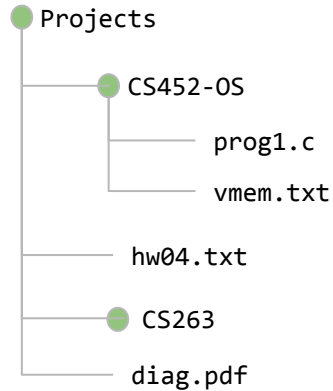
Filename	Inode Block
history.txt	9001735
two.pdf	1786

```
[17] ~$ ls -li history.txt
9001735 -rw---r-- 1 dulimarh users 12 Mar 22 2010 history.txt
[18] ~$ stat history.txt
File: history.txt
Size: 12          Blocks: 8          IO Block: 131072 regular file
Device: 34h/52d Inode: 9001735   Links: 1
Access: (0604/-rw---r--)  Uid: ( 2032/dulimarh)  Gid: ( 985/  users)
Access: 2018-08-22 00:30:36.775036674 -0400
Modify: 2010-03-22 16:43:02.000000000 -0400
Change: 2018-08-22 22:33:08.633587053 -0400
Birth: -
[19] ~$
```



Unix Directory Structure

Under Projects: 2 directories + 4 files = 6 inodes



DirEnt of Projects

.	_____
..	_____
CS452-0S	5781
hw04.txt	6234
CS263	_____
diag.pdf	8344

inode number

DirEnt of CS452-0S

.	_____
..	_____
prog1.c	4512
vmem.txt	7322

hard link count

inode

```

Home> ls -li
9322 drwxr-x--- 4 userone student 4096 Sep 29, 2008 Projects
Projects> ls -li
5781 drwxr-x--- 2 userone student 4096 Jan 17, 2013 CS452-0S
6234 -rw----- 1 userone student 17551 Oct 02, 2013 hw04.txt
7144 drw----- ? userone student 8192 Sep 14, 2013 CS263
8344 -rw-r--r-- 1 userone student 7366 May 22, 2013 diag.pdf
  
```

Inode 6234

-rw-----
userone
student
2013-10-02
17551
1
pointers to direct & indirect blocks

Inode 5781

drwxr-x---
userone
student
2013-01-17
4096
2
pointers to direct & indirect blocks

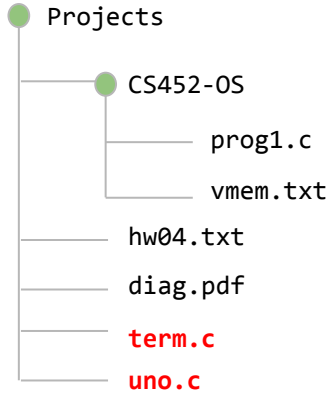


Unix File Linking

(no copying of file data)



Unix Hard Links vs. Symbolic Links



```
drwxr-x--- 2 userone student 4096 Jan 17, 2013 CS452-05
$ ln -s CS452-05/prog1.c uno.c # symbolic link
$ ln CS452-05/prog1.c term.c # hard link
```

Listing of Projects

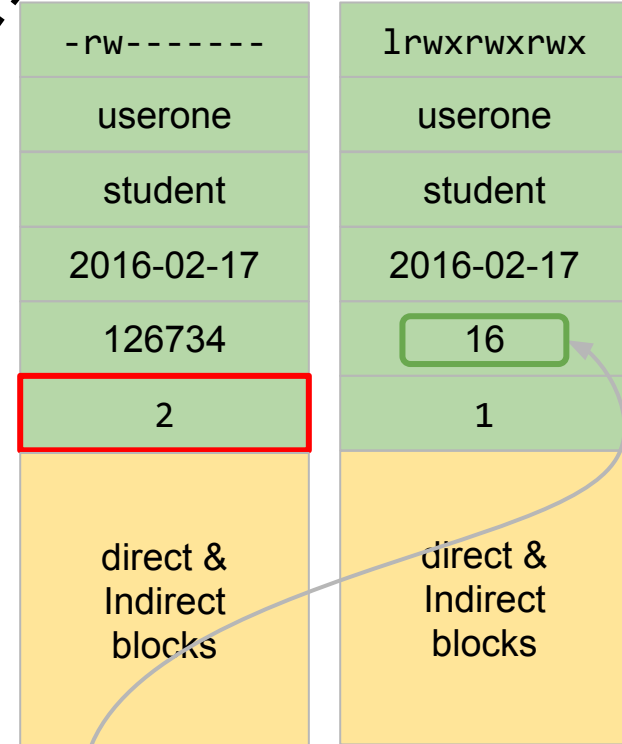
.	_____
..	_____
CS452-05	5781
hw04.txt	6234
diag.pdf	8344
uno.c	8881
term.c	4512

Listing of CS452-05

.	_____
..	_____
prog1.c	4512
vmem.txt	7722

Inode 4512

Inode 8881



CS452-05/prog1.c => 16 letters in filename

Hard Links or Soft Links?



`/usr/bin/firefox-latest inode# 3950`



`/apps/game/sudoku inode# 3950`

```
In /apps/game/sudoku /home/me/bin/sudoku # WON'T WORK  
In -s /apps/game/sudoku /home/me/bin/sudoku # OK
```


UFS File Format

- UFS (Unix File System) refers a set of FS variants based on Unix design
- **Boot block**: code to bootstrap the OS
- **Super block(s)**: metadata about the filesystem
 - Pointer to inode of the ROOT (/) directory
 - UFS: root inode id is 128
 - Linux Ext(2|3|4) root inode id is 2
 - Size of the entire filesystem
 - List of free inode blocks
 - List of free data blocks
- **Inode blocks** for storing index nodes (inodes)
 - One inode per file/directory
- **Many data blocks** (> 95% of the disk sectors)
 - Directory entry and File objects
 - [Journals used during recovery]



Important pedagogical concept:

Common *data structures* (list, trees, balanced trees, ...) can be implemented on a *persistent storage*





Pathname resolution:

`/path/to/my/file/somewhere.c`

Which inode?

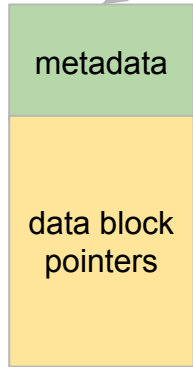


Resolving path /home/hans/hello.c

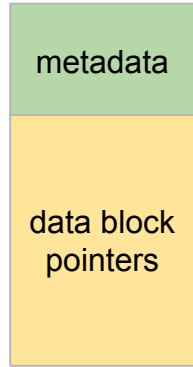


superblock

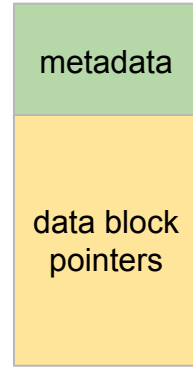
Linux Ext(2|3|4): ROOT inode is 2



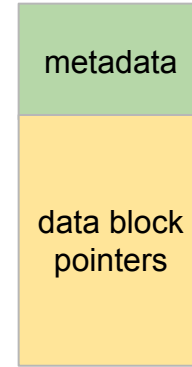
inode 128



inode 943



inode 446



inode 2921

inode blocks

.	128
..	128
home	943
lib	3379
usr	1172

root directory

.	943
..	128
beth	8457
flo	771
hans	466
jean	3379
mike	1172
nate	888

home directory

.	446
..	943
a.txt	6222
b.gif	318
hello.c	2921
p.doc	8833
p.pdf	9001
zero	888

hans personal dir

```
#include <stdio.h>

int main()
{
    printf ("hello");
    return 0;
}
```

hello.c

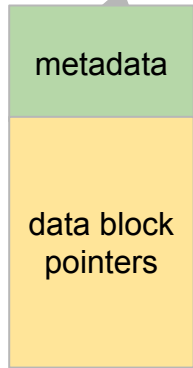
data blocks

Hardlink: ln /home/hans/hello.c /hi.c

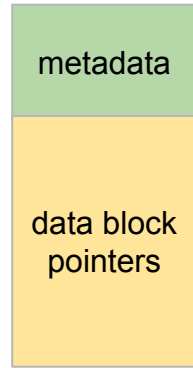


Linux Ext(2|3|4): ROOT inode is 2

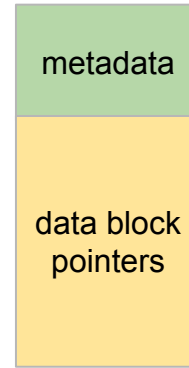
superblock



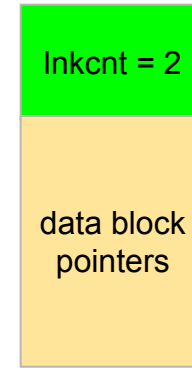
inode 128



inode 943



inode 446



inode 2921

inode blocks

.	128
..	128
home	943
hi.c	2921
lib	3379
usr	1172

root directory

.	943
..	128
beth	8457
flo	771
hans	466
jean	3379
mike	1172
nate	888

home directory

.	446
..	943
a.txt	6222
b.gif	318
hello.c	2921
p.doc	8833
p.pdf	9001
zero	888

hans personal dir

```
#include <stdio.h>

int main()
{
    printf("hello");
    return 0;
}
```

hello.c

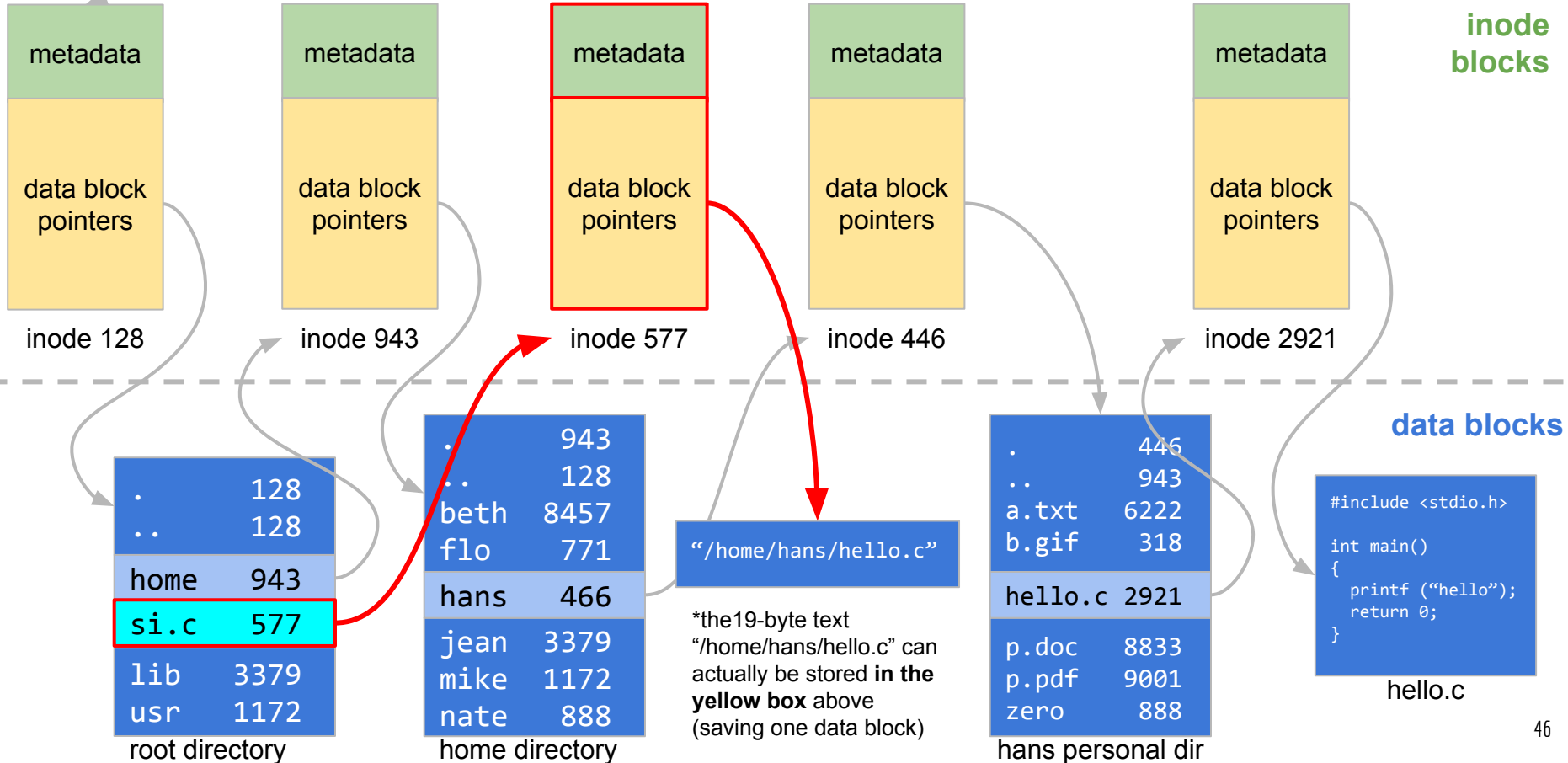
data blocks

In -s /home/hans/hello.c /si.c



Linux Ext(2|3|4): ROOT inode is 2

superblock



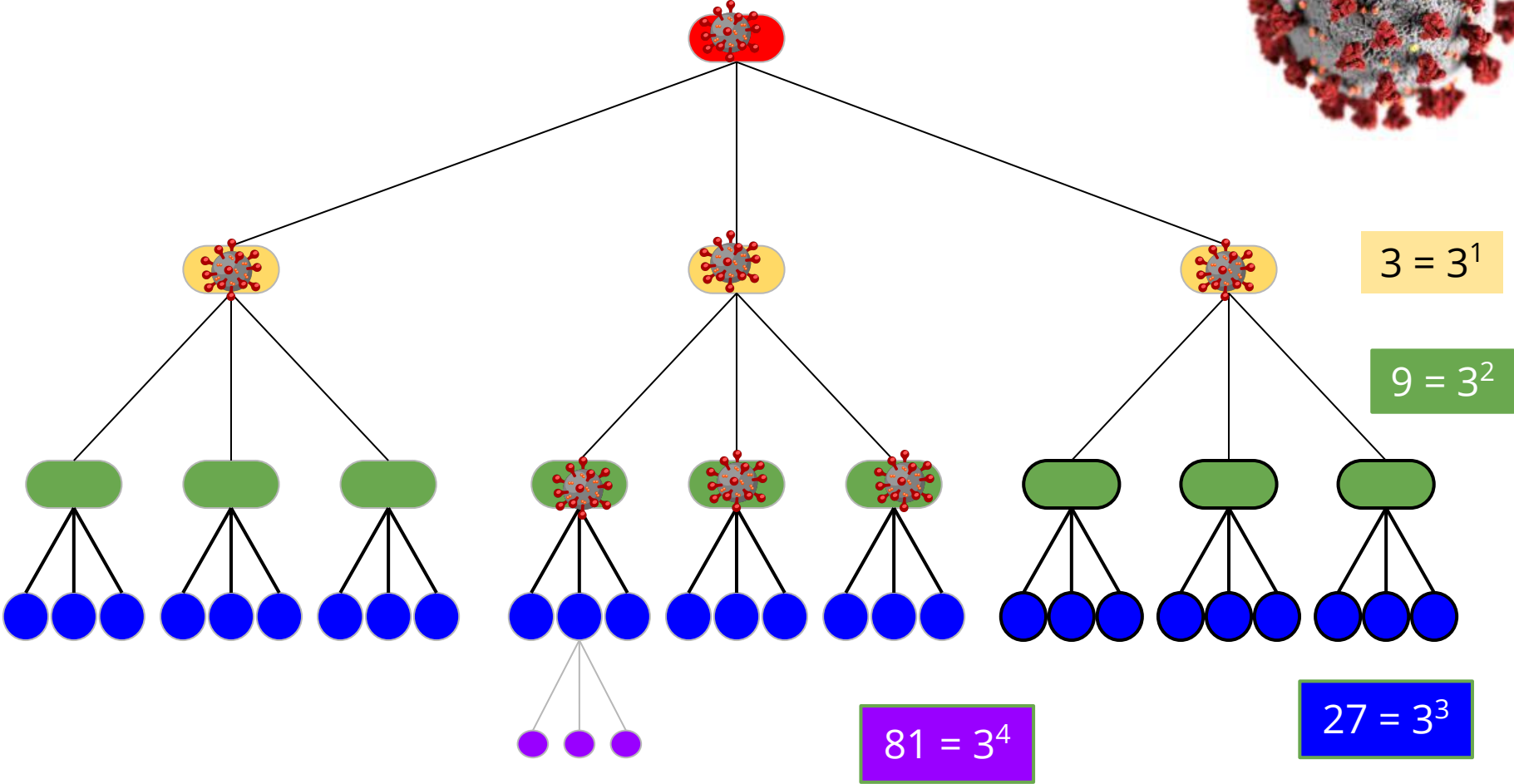
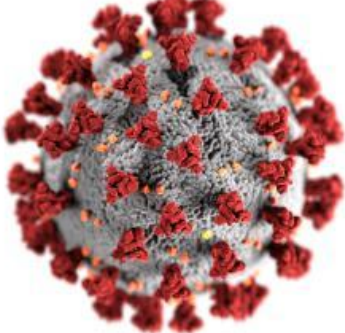
Linux/Unix File Allocation

- One inode per file / directory
- File block addresses are stored in inode and also in indirect blocks
- N (10 or 12) direct pointers and 3 indirect pointers
 - N direct pointers hold the addresses of the first N data blocks used by a file
 - The N+1st pointer holds the address of a single indirect block
 - The N+2nd pointer holds the address of a double indirect block
 - The N+3rd pointer holds the address of a triple indirect block

UFS Indirect Blocks

- Indirect blocks are **data blocks** used for storing pointers (block address) to data blocks
- Single indirect blocks
 - Contain pointers to data blocks
- Double indirect blocks
 - Contain pointers to single indirect blocks
- Triple indirect blocks
 - Contain pointers to double indirect blocks

Multi-Level Indexing



Unix: inodes and data blocks

Assume: blocksize 2K, 4-byte pointers
Therefore. 512 pointers per block

inode: 90+ bytes

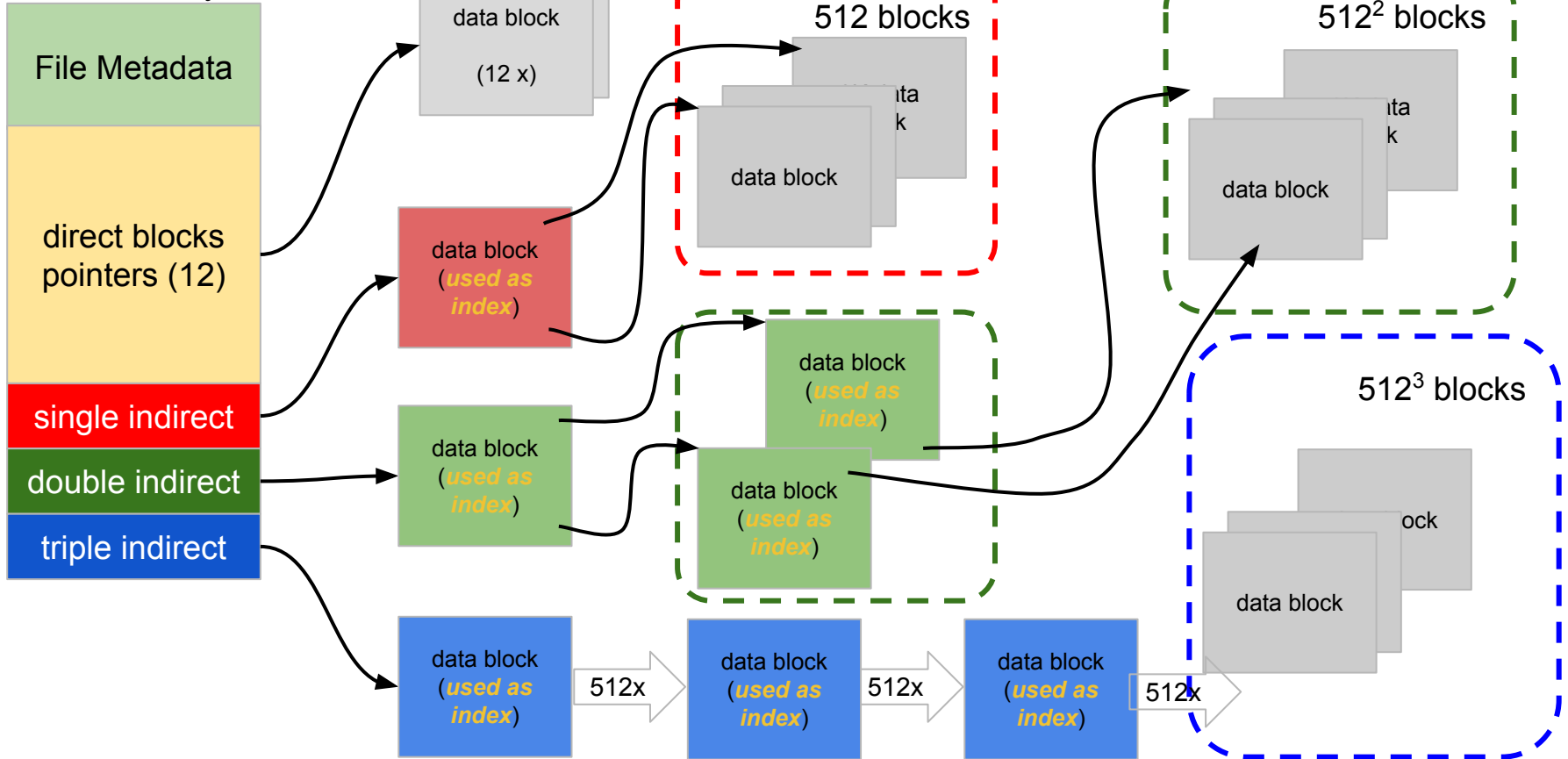




Diagram is not drawn proportionally. Inode is about 90+ bytes, data block can be 512 to 4096 bytes.

Max File/Directory Size in Linux/UFS

	Scenario #1	Scenario #2
Data Block Size	2K = 2048 bytes = 2K	4K = 4096 bytes = 2^{12}
Block Pointer Size	4 bytes	4 bytes
# Pointers per block	$2048 / 4 = 512 = 2^9$	$4096 / 4 = 1024 = 2^{10}$
<i>12 Direct Pointers</i>	$12 \times 2K = 24K$	$12 \times 2^{12} = 48K$
<i>Single Indirect Ptrs</i>	$512 \times 2K = 2^9 \times 2^{11} = 1M$	$1024 \times 2^{12} = 2^{22} = 4M$
<i>Double Indirect Ptrs</i>	$(512)^2 \times 2K = 2^{18} \times 2^{11} = 2^{29} = 512M$	$1024^2 \times 2^{12} = 2^{32} = 4G$
<i>Triple Indirect Ptrs</i>	$(512)^3 \times 2K = 2^{27} \times 2^{11} = 2^{38} = 128G$	$1024^3 \times 2^{12} = 2^{42} = 4PB$
Max Filesize	128G + 512M + 1M + 24K	4PB + 4GB + 4MB + 48K



FAT(12|16|32)



FAT32 Disk Layout

- Sector size = 512 bytes
- Reserved Sectors
 - Boot code
 - Partition Tables
 - Volume Identification sector
 - Bytes per sector
 - Sectors per cluster
 - Number of FATs
 - Sectors per FAT
 - First cluster of the ROOT directory
- FAT sectors
- Root directory
- Data sectors